



Gilbert, D. P., & Yearworth, M. (2016). Complexity in a Systems Engineering Organisation: An Empirical Case Study. *Systems Engineering*, 19(5), 422-435. <https://doi.org/10.1002/sys.21359>

Peer reviewed version

Link to published version (if available):  
[10.1002/sys.21359](https://doi.org/10.1002/sys.21359)

[Link to publication record in Explore Bristol Research](#)  
PDF-document

This is the author accepted manuscript (AAM). The final published version (version of record) is available online via Wiley at <http://onlinelibrary.wiley.com/doi/10.1002/sys.21359/abstract>. Please refer to any applicable terms of use of the publisher.

## University of Bristol - Explore Bristol Research

### General rights

This document is made available in accordance with publisher policies. Please cite only the published version using the reference above. Full terms of use are available:  
<http://www.bristol.ac.uk/red/research-policy/pure/user-guides/ebr-terms/>

# Complexity in a Systems Engineering Organisation: An Empirical Case Study

Dawn Gilbert. Systems Centre, University of Bristol  
Mike Yearworth. Systems Centre, University of Bristol

## 1. INTRODUCTION

The paper presents an exploratory case study carried out to address the research question ‘how does complexity in the organization affect the ability of Systems Engineers to meet delivery expectations in terms of cost and time?’ The review considers how literature in the Systems Engineering arena describes and addresses the problem of failing to meet delivery expectations. An absence of empirical research that considers a Systems Engineering organization as a complex adaptive system is identified.

The industrial context that led to this research is described, which, along with consideration of the literature review, supports the choice of methodology. Evidence is provided and analyzed from two perspectives; first a traditional reductionist view of business and project management within an organization, which is based on a deterministic relationship between cause and effect, and second a complex adaptive system view of the organization where cause and effect can be understood in hindsight, but not precisely predicted (Kurtz and Snowden, 2003). Within this perspective it is accepted that emergence is possible. This case study finds that delivery expectations are founded upon predictions of outcomes derived from deterministic causal relations. These expectations are not always met by the emergent outcomes that arise from the complex organization.

## 2. LITERATURE

### 2.1 Systems Engineering

Literature from the Systems Engineering domain was systematically reviewed. Key writings and authors on this topic were identified by searching within journals and proceedings from conferences that target Systems Engineers. Academic materials written by the key authors were also reviewed, as were sources that were frequently referenced. The material in the round recognizes that development projects may fail to meet delivery expectations, and that attempts to address the issue from within Systems Engineering are predominately positivist, relying only on quantitative methods (Gilbert, et. al, 2014). Muller et al (2013) describes organizational project management research as a field also dominated by reductionist paradigms and perspectives.

The dominance of *reductionist-by-default* research methods within Systems Engineering research is described by Brown (2009), who notes that Systems Engineering researchers are now moving beyond the realms of physical sciences, but wonders whether they are equipped to do so. Brown (2009) found in a review of selected Systems Engineering literature that methodology wasn’t always discussed, and when it was it was often describing methods used, rather than the logic underpinning the selection of the methods (i.e. methodology). This lead to a view that research across the industry draws ‘superficially (if at all) from any recognizable theoretical position’ (Brown, 2009).

Valerdi and Davidz (2007) note that ‘while many Systems Engineers have formal training in classical engineering disciplines, the problems faced in the field are both technical and social in nature’. A background in classical engineering, or physical sciences supports capability in positivist, reductionist approaches which are underpinned by a deterministic view of cause and effect. As problems faced by Systems Engineers span technical and social domains, the suitability of applying training from classical engineering within this new broader realm must be considered. They go on to describe challenges of empirical research in Systems Engineering, citing the relative immaturity of the field, the lack of appreciation for empirical research, the lack of access to data, and a lack of accepted metrics. They also note that practitioners training ‘often emphasizes a positivist, quantitative approach to research inquiry’ in calling for case study research as ‘a constructive way to deal with the non-reductionist form of SE required for dealing with complex systems’. Valerdi et al (2010) push the Systems Engineering research community to consider hybrid approaches within a philosophically consistent framework. This literature provides some level of explanation of why approaches taken by the Systems Engineering community to understand the issue of failure to meet delivery expectations are primarily reductionist and deterministic in nature.

The ongoing research effort within Thales UK explores the influence that complexity may have on the ability to meet delivery expectations with the aim to reduce or eliminate ‘problematic projects’, which are projects where expected costs are exceeded by more than £10million, and the cost overrun is not anticipated by high level management. The particular aspect of complexity under consideration is where the relationship between cause and effect can only be seen in retrospect (Kurtz and Snowden, 2003; Snowden and Boone, 2007). The term ‘complexity’ has no singularly agreed definition (which is not an issue unique to Systems Engineering), and while Systems Engineers may refer to complexity frequently, the precision with which the term is used in literature and communication varies. This issue was encountered by Sheard (2012) in a retrospective survey to study the impact of complexity variable values on Systems Engineering projects. In an earlier paper, which supported this later work Sheard (2009) provides a definition of a complex system (which does not conflict with some of the many definitions of a complex adaptive system), and considers the INCOSE organization as an example. This theoretical view of a particular organization that Systems Engineering practitioners may be familiar with as a complex adaptive system is the singular example that could be found in Systems Engineering literature that considers an organization as a complex system.

## **2.2 Organizations as Complex Adaptive Systems (CAS)**

A CAS evidence scan (The Evidence Centre, 2010) finds that CAS work in the engineering sector is mostly descriptive or theoretical, rather than empirical or applied. It also identifies CAS thinking and approaches in use in management, urban planning, economics, environmental science, education, and healthcare. Gell-Mann (1994) states the term CAS has different meanings to different people, this view is also supported by the findings of the CAS evidence scan, which notes the most common definition of a complex adaptive system is based on the work of Holland (The Evidence Centre, 2010). The research presented in this paper thus uses Holland’s (2014) definition of CAS applied to an organization where the staff are the agents of a complex adaptive system. In particular Holland’s description of the activities of agents at a granular level of detail is applied (Holland, 2014 pp26-32). It states that all CAS agents (which learn or adapt in response to interactions with other agents) have three levels of activity; 1) performance of moment-to-moment capabilities which designate an agents behavioral repertoire at a point in time, 2) credit-assignment whereby the usefulness of available capabilities can be rated higher (than others) as experience finds that those capabilities lead to rewards, and 3) rule-discovery whereby the agent generates new capabilities. Holland also notes that the study of CAS is still in its early stages, so only pieces of a theory exist (ibid).

Palmberg (2009a) puts forward a tentative model for managing organizations as CAS in work that contributes to her doctoral thesis (Palmberg, 2009b), which also includes a case study methodology for exploratory and interpretive empirical research in this field, which considers the work of (Yin, 2014). Friedman and Sage (2004) discuss the role of case study research in Systems Engineering and provide an example case study framework for Systems Engineering and systems management, also derived from (Yin, 2014).

The systems science community which joins the Systems Engineering community through the ISSS and links with INCOSE have published empirical research that views a project team as a complex adaptive system to explore the relationship between group development and complex adaptive cycles (Edson, 2012). The research is a retrospective revelatory case study, using Holland’s definition of CAS. Palmberg (2009b) articulates a research methodology for exploratory empirical research of organizations as CAS, but no empirical research of a Thales-like organization that develops complex products could be found in this review.

## **2.3 Ontological and Epistemological views of Organizational Change**

Van de Ven and Poole (2005, p1381) present a typology of four approaches for studying organizational change. They are defined by one of two ontological views (whether organizations consist of things, or processes) and by one of two epistemologies, one that ‘seeks explanation of deterministic causation’ (referred to as variance methods), and one which provides ‘explanations that tell a narrative or story about how a sequence of events unfolds to produce a given outcome’ (referred to as process methods). Van de Ven and Poole note that the four approaches may be viewed as opposing or competing views, though the authors see them as complimentary, arguing that pluralist insights provide a richer understanding.

Within Systems Engineering literature, Valerdi et al (2010) call for hybrid approaches in Systems Engineering

research, and Midgley (2011) makes a case for theoretical pluralism partnered with methodological pluralism. The Van de Ven and Poole (2005) typology provides an evaluative framework for considering different perspectives on organizational change.

## **2.4 Contribution of this Paper**

The Systems Engineering literature reviewed highlights a gap, but also an appetite for theoretically rigorous empirical research that considers an organization as a CAS, and investigates how this might shed light on the issue of project delivery performance across a Systems Engineering development organization.

This paper brings together these arguments and gaps to present a single revelatory empirical case study with multiple units of analysis considered from two perspectives i) the ‘traditional’ Systems Engineering perspective, which considers an organization to be comprised of things and considers change to be driven deterministically and ii) a ‘Complex Adaptive Systems’ view which captures process narratives and identifies and expects emergence.

## **3. INDUSTRIAL CONTEXT**

Thales UK is comprised of five operating domains, each of which is comprised of a number of business lines. Each business line carries a portfolio of projects that are, at any one point in time, in varying states of completion. The first author is a systems researcher, embedded within Thales UK and affiliated with the corporate Systems Engineering function, rather than with any one domain, business line, or project. The systems researcher was placed within the case study business line at the request of the business line to help them understand more clearly whether their work was likely to meet solution delivery expectations in terms of timeframe and cost. An embedded research position provides a rare opportunity to carry out and publish empirical research of a Systems Engineering development organization, and so provides the context to carry out a revelatory case study. Systems engineering practitioners rarely have the combination of research skills, non-disclosure agreements, and available time to perform and publish this kind of research. Conversely, academic researchers rarely have the access, and ability to appear as ‘an insider’ that is needed to carry out and publish empirical research from within a Systems Engineering organization.

### **3.1 Organization of the Case Study Business Line**

Nominally, Thales UK is a matrix organization (Hobday, 2000), with a dominating project structure and weaker functional structure as shown in Figure 1. The Project Manager (a member of the Program Management Function) reports to the Business Line Management through the dominant project reporting line, and to the Program Management Head of Function through the weaker functional reporting line. Similarly, Systems Engineers report to the Project Manager through the project reporting line and the Systems Engineering Head of Function through the functional reporting line. As systems under development are decomposed to sub-systems, which include software, software engineers report to Systems Engineers in the project line. Software engineers report to the Software Engineering Head of Function through the functional reporting line. Thales Group has an enterprise architecture, which describes tailorable processes, methods, and tools that should, or may be, used within business lines and other corporate entities. This architecture ensures coherence between projects, function, and management roles within Business Lines and the wider organization.

In the case study business line, and within the boundaries applied to the case study dataset, individual projects are related to one of three products, each of which include major software sub-systems. Staff state that software is developed using an agile scrum methodology, Systems Engineers describe their lifecycle as vee, or agile front-end and vee back-end.

“Software is running agile, systems is still traditional” (Key Stakeholder, Head of Systems Engineering)

“Software is agile, systems is v” (Head of Software Engineering)

“It’s agile for the first 2/3<sup>rd</sup> of the v, then systems IVVQ for the last part” (Scrum Master – Product 2)

The nature of the organization provided an opportunity to explore how complexity affects the ability of Systems Engineers to meet delivery expectations by examining a number of units of analysis concurrently.

## 4. METHODOLOGY

The research stance is critical realist (Sayer, 2000; Mingers 2004; Bhaskar 2008). The empirical and socio-technical nature of the case study means evidence gathered from within the organization may be first order data (e.g. a number in a spreadsheet, a statement made in conversation) or second order data (e.g. evidence of how first order data is *interpreted and used* by staff in the case study organization). This requires a capture of multiple perspectives, which can be appreciated by the stratified ontology of critical realism. Some staffs believe that nothing exists beyond the empirical world, and enact management approaches that align with this view. Others believe that aspects of the world exist beyond our ability to know and experience them, and similarly, enact management approaches that align with this view. Critical realism is also concerned with causation, and provides a framework through which both deterministic causality – where a specific cause leads to a specific effect with regularity – and emergence – where cause and effect can be linked in hindsight but can not predicted with certainty – can be understood. Causality is of interest because it is a mechanism that could be leveraged by management to realize improved performance.

### 4.1 Analytical Framework

Our analytical framework considers two perspectives.

The first is a ‘traditional’ epistemological and ontological perspective, which is prevalent in project management and Systems Engineering research. Implicit in the emphasis on ‘positivist, quantitative approaches to research inquiry’ identified by Martin and Davidz (2007) is a preference for an ontological view of an organization that consists of things and an epistemology based on deterministic causation. This perspective leads to a focus on planning and plans that describe quite specifically how the future will evolve, with the expectation that our view of the future will materialize as planned; i.e. we have a belief that patterns of cause and effect can be manipulated through managed control. Van de Ven and Poole (2005) describe this perspective as ‘approach 1’. This view also encapsulates the interpretations of complexity that consider that the future can be precisely predicted, and therefore managed and controlled, if we build the correct model from which to perform the predictive analysis. This view aligns with the right hand side of the Cynefin Framework (Kurtz and Snowden, 2003; Snowden and Boone, 2007).

The second perspective considered within the analytical framework views the organization as a ‘Complex Adaptive System’ comprised of Agents, which in this case are considered to be the staff in the organization. This perspective is described by Van de Ven and Poole as ‘approach 2’ and ‘approach 3’, which understand change through a process narrative. The process narrative is captured as part of the data collection stage; it reports the beliefs, understanding, and motivations of the staff that, through agency, act within the organization. In this view, outcomes are emergent, rather than realized through planning, prediction, and control. Staff are agents within the system exhibiting free will; they are neither precisely predictable, nor is the range of actions they may take knowable in advance, at individual or group levels. This view aligns with the left hand side of the Cynefin Framework (Kurtz and Snowden, 2003; Snowden and Boone, 2003).

### 4.2 Method

A participant-observation approach was used in field-work, where data was collected for consideration in real-time, and for in-depth analysis at a later stage. Data collection progressed in consideration of ethical dilemmas and tensions between seeking rigor and establishing and maintaining trust within the staff group. Stakeholder engagement was primarily 1:1 as a mechanism to develop greater trust and elicit more honest and open views from interviewees. All staff engaged with were made aware that the systems researcher was carrying out research, and that appropriate security clearances and Non-Disclosure Agreements were in place to protect sensitive data and systems, yet still provide access and allow publication. Semi-structured interviews were conducted in a variety of ways ranging from formal meetings to informal chats. They used the vernacular of the business line and mirrored the tone of the interviewee. They varied from being specifically planned, and scheduled by email and held in meeting rooms, to being more opportunistic in nature, for example by triggering a conversation in the kitchen while making a cup of coffee, or by a staff member dropping by the system researcher’s desk for an unprompted chat. Questions were open-ended, asking why, how, when, etc. in order to capture an expression of the interviewees beliefs, working

methods, and approaches, and their understanding of causal relationships, and also to obtain sources of verification or triangulation data to compare to evidence already captured from that interviewee, or from other sources in the business line. Archival and quantitative data was accessed via standard industry tools for finance, schedule, and resource profiling (Primavera®), task and problem tracking (Jira®) and System requirements management (Doors®). Other soft copy data was accessed via the business line network drives (as a non-project worker, the system researcher's network access was restricted to management and project meta-data), and the corporate intranet.

### **4.3 The Scope, Processing, and Consolidation of Data**

Data was collected from the case study business line over a 6-month period and amalgamated in to a case study database. The case study database includes soft-copy reports, archival records, a transcript of one audio-recorded semi-structured interview, and ethnographic notebook entries which captured the system researcher's actions, short-term reflections, and plans as well as recording direct observations, and notes from meetings, discussions, and telephone calls. Soft copies of email correspondence were also captured.

Boundaries were applied to the dataset to progress the work toward a discrete piece of research that could be reported concisely, yet still be substantive. The boundaries selected were:

- The case study business line. The case study business line interacts minimally with its domain and other Thales UK business units.
- Projects relating to the three main products. The bulk of the engineering headcount in the business unit is committed to developing and supporting three main, predominately software-based systems.
- The key stakeholder and individuals I was subsequently referred on to either voluntarily, or through my own exploration of key relationships.

Applying these boundaries added a level of manageability to the case study database, yet still provided ample data to consider the research question that seeks to understand how complexity in the organization affects the ability of Systems Engineers to meet delivery expectations.

The application of these boundaries reduced the dataset from notes of 72 meetings to notes from 43 meetings with 12 staff, which were typed up from handwritten notes, resulting in a 44-page 21,000 word report which references 63 staff members, 7 business processes, 6 work sites, 4 commercial organizations, 4 customers, and 4 Thales entities, all of which have been anonymized. Note, the original case study database still exists and is available to allow broader and deeper analysis in support of the ongoing research program. Extensive raw data is not presented in this paper to retain confidentiality, though representative samples are provided.

## **5. EVIDENCE AND ANALYSIS**

Evidence relating to three topics was extracted from the case study database, and viewed through the two perspectives considered by this research. The themes are agile scrum sprints, staff circulation, and management databases. The units of analysis are the business line, the topic, and the individual stakeholders themselves (Yin, 2014, p54). Structuring the evidence and analysis in this way allows us to reveal how evidence within the units of analysis help address the research question.

The 'traditional' perspective applied across the three topics would allow the aims and objectives of using agile sprints, a staff circulation policy, and a collection of management databases to be identified, and would allow deterministic causal evidence to be identified that shows the aims and objectives are satisfied by the structures that are in place. This 'traditional' view assumes all individuals have a common understanding of the aims and objectives of the business line, and they act uniformly and consistently to pursue these aims, since it is based upon the premise of deterministic cause and effect.

The 'Complex Adaptive System' view provides a narrative that describes the evolution of events during the 6-month investigation. As such, the data themselves reflect occurrences in the organization within the specific time period of data collection. The organization, as represented by the narrative data, existed at the time but has subsequently continued to evolve and is now a different organization when considered at a detailed level. In this way, the data themselves become obsolete, though the nature of the dynamics observed within this case study may

persevere, and indeed may be familiar and recognizable to practitioners working in other Systems Engineering organizations.

## **5.1 Topic 1 - Agile Scrum Sprints**

The Systems Engineering development approach includes an architectural decomposition of the solution in to its component sub-systems. The three products under consideration in this Business Line each include major software sub-systems. The prevalence of a reductionist paradigm within the organization leads practitioners to conclude that, if the software sub-system team fails to meet delivery expectations (where the software team deliver a key subsystem to the Systems Engineering team, who then integrate sub-systems to make the whole), then the Systems Engineering team, as a knock-on effect, may subsequently fail to meet delivery expectations. This relationship illustrates how the ability to meet delivery expectations for software development is related to the ability of Systems Engineers to meet solution-level delivery expectations. The review of project development progress across the portfolio of all active projects across the business line is described in section 5.1.3.

The software sub-systems developed in the Business Line are developed using agile approaches, specifically Scrum. ‘Scrum is a process framework that has been used to manage complex product development since the early 1990’s. Scrum is not a process or a technique for building products; rather it is a framework within which you can employ various processes and techniques’ (Schwaber & Sutherland, 2013). A Scrum Team develops products iteratively and incrementally. The Scrum Master helps those outside the Scrum Team understand how they can most productively work with the Scrum Team. ‘The heart of Scrum is a Sprint, a time-box of one month or less during which a “Done”, useable, and potentially releasable product Increment is created. Sprints best have consistent durations throughout a development effort.’ (Schwaber & Sutherland, 2013). Scrum theory, practices and rules require consecutive sprints to comprise the same Scrum team members, each of whom would be fully committed to the sprint, and not carrying out responsibilities elsewhere in the organisation. Each sprint is planned up-front, with work to-be-completed represented in the sprint backlog. The sprint backlog remains stable throughout the duration of the sprint. These consistencies would lead to improved sprint ‘velocity’ (a term relating to productivity) over time. Sprint velocity is expected to improve as the group, working together, becomes more effective and therefore more efficient over time. The velocity, and amount of work that was planned and actually delivered during each sprint, can be viewed in a standard 7-sprint view in Jira. The rate of increase in velocity can be monitored to inform planning in upcoming sprints.

### ***5.1.1 The Traditional Perspective***

Jira is used to track tasks and problems in the software products under development, and as administrative support for agile sprints. Through a deterministic causality perspective, a sprint would be planned, its execution would follow the plan and the committed work would match the completed work, unless errors were made.

The committed (planned) work and completed (actual) work were compared for seven consecutive sprints for each of the three products by examining the standard Velocity Charts from Jira. Each product is developed by multiple sprint teams, each with their own sprint backlog to work through during each sprint. The amount of planned and actual work contained in each sprint backlog is expressed in quantitative units, allowing statistical analysis of the results.

Plotting planned against actuals should yield a scatter plot with line of best fit described by  $m \rightarrow 1.0$ , and  $r^2 \rightarrow 1.0$  for large sample sizes.

A total of 91 sprint results were gathered across the three products (product 1 provided 42 data points, product 2 provided 21 data points, product 3 provided 28 data points). The r-squared of all 91 points was 0.92, and its line of best fit is described by  $y=0.68x$  (forcing a 0,0 intercept). See Tables 1 to 4.

This analysis of the traditional perspective confirms the sprint is not performing as planned, or as expected by this perspective. This view is unable to confirm why performance is not as expected.

### ***5.1.2 The Complex Adaptive System Perspective***

The study database is examined to consider evidence of beliefs and perspectives of how Jira and the agile scrum are used, and how this helps meet delivery expectations.

“The project has 3 pulls..the PM line...through operations about cost...and to the design criteria” (Systems Engineer – Product 3)

“What we struggle to do is estimate and stick to it. Across a number of projects it’s pretty much the same year after year in the software domain...One cause of problems is other activities like business winning that’s over and above, but knowing that, we said we’d do the planned work” (Program Manager – Product 2)

“The program manager asks why we aren’t loading [staff] to 100%...when reporting time in timesheets people put flat time across each day, we encourage Jira time reporting to be accurate...The project team changes, 1700 hours max at 18 engineers, 300 hours low, 400 hours is 6 engineers. I draw a line around staff time. The program manager asks ‘why does it take so long?’ I’m trying to do an estimating process that tells the story that the PMs will want. (Scrum Master – Product 1)

“The main risk is in software development....The business objective here, I want quantitative and predictable software performance, delivery against budget” (Program Management Head of Function).

“Reports are frozen with history, containing planned and actuals...The asterisk means added to the sprint after the start, if it goes in, its urgent, it can come out if it’s not needed, something else changes, or there’s no time... Velocity doesn’t work for us, it assumes we have a constant team, and the team doesn’t work on the sprint work 100% of the time” (Scrum Master – Product 1)

Consideration of this sample of narrative data shows that individuals may desire predictable software development performance, and may recognize competing interests that are managed tactically on a day-to-day basis. Scrum Master – Product 1 tries to tell a story, in recognizing that a narrative reveals the dynamics that are in effect in managing the software development process, but the stakeholders who would benefit from understanding the narrative (Program Management Head of Function and Program Manager – Product 2) are not open to non-quantitative narrative forms. Their dominant paradigm cannot appreciate the complexity and dynamics that the scrum master is attempting to manage.

Through semi-structured interviews and email follow-up, Scrum Master – Product 1 was able to provided evidence that underpinned his view. The data quantifies the effects of some aspects of ‘reality’ that do not conform to the scrum methodology approach, specifically, in practice the sprint team is not consistent, their time is not 100% dedicated to the sprint (Table 5), and the backlog does not remain stable during the sprint (Table 6). This is a result of the short-term demands on staff that are affected through the organizational structure, which at a minimum asks staff to satisfy demands from both the project and functional directions. Table 5 and Table 6 provide data that report the magnitude of the key drivers that Scrum Master – Product 1, believes most strongly influences the ability of the sprint-team to deliver the backlog as planned each sprint. Table 5 shows the equivalent staff utilization of a Product-1 sprint team over 5 chronological sprints. The sprint team grows from 8 to 12 people, and utilization across an entire sprint varies from 100% to 29%.

Table 6 shows the magnitude of unplanned staff absences, work removed from the sprint during the sprint, and work added to the sprint during the sprint. Abortive work is work completed in the sprint on tasks that were not finished and were removed from the sprint mid-sprint. The magnitude of work delivered, compared to the magnitude that was planned across these seven sprints varies from around 80% of planned to 110% of planned. Average sprint churn (new tasks as a percentage of originally planned tasks) is 21%.

Table 5 and Table 6 provide quantitative data that apparently disproves the validity of the dominant traditional approach to managing. Though Scrum Master – Product 1 could clearly articulate the dynamics that led to emergent performance of the sprint team, they struggled to pass this understanding on to other managers.

### ***5.1.3 The Business Line Level***



The Thales Organizational Management System states that monthly Program Review Meetings will be held and they are a forum where issues affecting technical development relating to delivery are evidenced and discussed, supported by an up-to-date Program Review Pack. This mechanism provides a forum to present and discuss the data in Table 5 and Table 6, and its origins and implications with engineering and project management functional leaders in the Business Line. It is mandatory for the Program Management Head of Function and the Operations Director (who at the time was also the Systems Engineering Head of Function) to attend the meetings. The meetings are a forum through which Business Line management monitor the active portfolio of programs, and provide strategic direction, support, and feedback to the project teams. The following quotes were provided by interviewees chronologically.

“We look at the Program Review Pack and we don’t get good information on the state of the project....PMs are reluctant to step out of the standards scripts of headings for headlines” (Program Management Head of Function).

“The program review packs still don’t get looked at, they don’t look at my supplementary pack [which reports sprint performance, lessons learned, engineering metrics, context, achievements, and upcoming milestones] “ (Scrum Master – Product 1)

“I think from a business perspective we need, what we needed was some relatively simple metrics that could be collected easily” (Key Stakeholder - Acting Business Line Operations Director, and Head of Systems Engineering Function)

The Acting Business Line Operations Director, who was also Head of the Systems Engineering Function (and the Key Stakeholder in this research) believed, at that point in time, that evidence relating to the ability of project engineering teams to meet delivery expectations was not available as part of the Program Review Pack, and therefore could not inform the discussions at the review meetings. In using the specific case of Product 1 to challenge this view, the Key Stakeholder stated that they had not actually attended recent meetings.

The attendance of the Key Stakeholder in the program review meetings is stated as mandatory in the business management collateral as the Business Line Operations Director, but not as the Business Line Systems Engineering Head of Function. Program Review Meetings for the Business Line are held over two days each month. During the 6-months placement when this fieldwork was being performed, the Key Stakeholder attended all Program Reviews during two months, none of the program reviews during two months, and one-of-two days of program reviews during the other two months. During the last month of the fieldwork, a separate permanent Operations Director was appointed and in place. The permanent Operations Director did attend the program review meetings, as is stated as mandatory for that role. At the time, there was no other formal forum required by the Organizational Management System for the Systems Engineering Head of Function to engage with Project Engineers to understand progress and pressures.

#### ***5.1.4 Traditional and Complex Adaptive Systems Perspectives***

The dynamics affecting the ability of the scrum to meet delivery expectations are not appreciated by a traditional view of the organization. The Scrum Master takes steps to inform management of discrepancies between planned and emergent achievements and their causes. The project and business line management show a preference to pursue performance that is predictable and they continue to deliver a prescriptive plan, even in the face of evidence that the context is too turbulent to achieve the desired performance levels using this approach. A CAS perspective understands this behavior as strong values being attributed to reductionist approaches (credit assignment), to the extent that rule discovery, and the generating of new capabilities (an appreciation of the complex dynamics in effect within the sprint team) does not occur as performance (moment to moment capabilities) is directed away from appreciating information that challenges the dominant paradigm.

The Key Stakeholder, who was the Business Line Head of Systems Engineering, while also acting up in the Business Line Head of Operations role was not present at meetings their attendance is mandatory in. A traditional management perspective offers no explanation as to how staff will or should prioritize their efforts when they are overburdened with work. A CAS perspective expects agents to prioritize moment-to-moment according to their own personal, locally rational, view of what activity will be most beneficial. The CAS view can therefore explain why a staff member performing two roles simultaneously may not attend a mandatory meeting. A CAS perspective

accommodates the evidence that the stakeholders involved have different and persistent perspectives. CAS dynamics are evident in this data, and they affect the ability to meet delivery expectations, however, the management personnel show no evidence of attempting to understand or manage the CAS dynamics.

## **5.2 Topic 2 - Staff Circulation**

A theme of staff circulation affecting ability to complete work as planned arose through the stakeholder engagements.

### **5.2.1 The Traditional Perspective**

The corporate intranet and business line management collateral that was provided by stakeholders, or accessible on the local network drive, describes the 1-year business plan for the business line, and shows how that supports longer term strategies and higher level business objectives for the business line and the parent Thales domain which are articulated in other business planning documents.

The business plan for the Business Line identifies financial turnover and profit targets for the year, along with a detailed implementation plan, which includes a staff rotation target. The relevant sub-heading plans to develop a high performance team culture within the business line by ‘optimizing and enhancing capability through personal development and 10% staff rotation’. The plan states that feedback would be gathered from managers and staff, quarter-by-quarter, to continuously improve the approach. In practice, however, the feedback was not gathered. These documents and statements reflect a traditional perspective in business planning as cause and effect are identified, along with an implementation plan that will ensure that the objectives are delivered.

A sample from the Case Study database relating to staff circulation is provided by the following quotes:

“We are rotating staff left right and center so we have program mobility....The last 2 sprints have had to pay significantly for new people. You don’t get it and you won’t get it for several months, this is the rotation plan....” (Scrum Master - Product 2)

“There is an administrative overload related to juggling tasks so people who have fewer skills are occupied” (Scrum Master – Product 1)

“We’ve had people coming in and out of the project, agile should keep the team together..” (Systems Engineering – Product 3)

“We shuffle resources around, attempting to achieve an increase in knowledge capability and more interest, but we don’t know the impacts...The rotation aim is 10% per year...There is much greater than 10% circulation going on” (Program Manager – Product 2)

The individual stakeholders describe either the effects of staff rotation, or the aims of staff rotation, but no one person had a view that covered both. Staff rotation was described as impacting the ability to efficiently and effectively develop software, which in turn impacts the ability of Systems Engineers to manage the overall development lifecycle. Systems engineers described staff rotations, but did not note an impact to the ability to perform Systems Engineering work. Of the 12 staff who directly contributed to the bounded case study database examined here, 4 changed permanent roles during the 6 month period of embedded research, with a further 4 changing roles in the 6 month period that followed, giving an annual staff churn of 66% in this sample.

### **5.2.2 The Complex Adaptive System Perspective**

The systems researcher then contextualized locally gathered data within the system dynamics model described in Van Oorschot, et al (2010, p831) relating to assimilation delays and reallocation delays, and presented it to the Key Stakeholder. The perspective provided by this aspect of the system dynamics model shows how overall productivity is decreased if staff rotations occur faster than novices can progress to expert levels when assigned to new types of work. This approach stretched the Key Stakeholder to consider nonlinear dynamics within the organization, as a departure from a more traditional mechanistic view.

“Ah, it might be that we’ve had a worse impact than we thought, that’s enlightening, we want to cope with the perturbation...maybe we are a more competent engineering organization than we thought” (Key Stakeholder, Acting Operations Director and Head of Systems Engineering Function)

The engagement with the Key Stakeholder had revealed to them a tension between competing ideals – circulating staff to increase overall capability while also retaining experienced staff to deliver development work efficiently. The systems researcher used this approach to engage with the key stakeholder as it responded to the initial need for the placement in the Business Line, and it also provided a mechanism to expose the key stakeholder, when viewed as an agent in a CAS, to novel information and constructs. This allowed the possibility to observe the response of the Key Stakeholder over time.

During the 6 weeks that followed this discussion, this same grounded approach was used again to help the Key Stakeholder understand dynamics and emergent properties of other variables within the Business Line engineering function. This allowed potentially beneficial interventions that were available to the Key Stakeholder to be identified.

“I like that....you know what, I’d be quite keen to try and do something like this on something like Product\_4” (Key Stakeholder, Acting Operations Director and Head of Systems Engineering Function)

Viewing this response using a Complex Adaptive System perspective, the Key Stakeholder (agent) had discovered the value of an alternative method of understanding what affects the performance of the engineering function (rule discovery). Considering the options available to them, they considered an approach based on complex dynamics to be of higher value than the traditional reductionist planning and control techniques (credit-assignment), to the point where the way in which the Key Stakeholder planned to allocate their effort would now change (performance).

While the Key Stakeholder did not actively seek alternative perspectives on how the performance of the Business Line engineering function could be understood, an adaptation in response to the systems researcher’s actions (presentation of dynamic relationships, rather than the quantitative performance data that was expected) led to an alternative perspective being considered, found valuable, and acted upon.

### **5.3 Topic 3 - Management Databases**

Operations within the business line are expected to be carried out in line with the process-based Organizational Management System. Various management systems and tools are used to support this; of relevance to managing Systems Engineering development projects are Business Planning, Primavera, and Jira. Each of these tools includes their own database. This analysis explores the espoused contents and ostensible uses of the databases, and the beliefs held by the staff about their purpose, content, and use from a reductionist and CAS perspective. Table 7 outlines the three tool databases and gives a headline view of what the databases contain.

The following quotes illustrate the beliefs held by the staff about the tools.

“When you look in Jira, make sure what you are looking at is what you are thinking you are looking at” (Scrum Master – Product 1)

“In actuals, is overtime visible or not? – it’s not visible on actuals...we are mixing apples and pears, we are mixing the contents...” (Systems Engineer – Product 3)

“The cost collection is actually at different levels, for example, from subcontractors and from Jira. The lowest level we go should be the level you get actuals against. Primavera breakdown was mapped against delivery structure routed through work packages by customer, but we estimate and collect cost through product and function.” (Project Controller – Multiple Projects)

“A Thales UK Corporate Business Planning Lead was hard on prescriptive direction, now it’s ‘you tune the process for what works for you’, we’ve gone full circle back with everyone being in the room at the

same time....You've got to have human beings in the loop" (Key Stakeholder, Acting Operations Director, and Head of Systems Engineering Function)

"We want to integrate Jira to Primavera for work package management and earned value management. I already look at software burndown daily" (Head of Software Engineering Function)

"We've been trying to sort out how we do agile...last year we had a workshop on linking Primavera to Jira...Primavera tells you how many people you need in the sprint to do it. The actuals should be in Jira and Primavera...we export from Jira back to Primavera as target EVM" (Program Management Head of Function)

The staff, as a collective, consider the data held within the suite of databases to be of varying quality. There are conflicting reports of how the data is used, for example, whether the Primavera schedule defines the size of the sprint team, or whether the sprint team is fixed and does the work that it can.

Scrum Master – Product 1 and Systems Engineer – Product 3 identify that data in the databases are not actually what they might appear to be. The Software and Program Management Functional Managers use the data in the databases to make decisions. This is characteristic of a reductionist perspective, where the data in the database is considered to be objectively representative of 'the world', such that decisions can be made without need for case-specific data validation.

The Software and Program Management Functional Managers describe a desire to link the databases, or belief that it actually already occurs. The Project Controller notes that Primavera and Jira are not structurally similar enough to be linked currently (a claim that was validated by independently acquired data).

The Key Stakeholder notes that *"you've got to have human beings in the loop"* for work they personally participate in, but when considering the work being carried out on projects, appears to initially assume reductionist, impersonal approaches are adequate *"what we needed was some relatively simple metrics that could be collected easily"* (Key Stakeholder).

### **5.3.1 The Traditional Perspective**

A traditional reductionist view of organizational management assumes that quantitative data represents an objective reality, and that the data would be understood comparably by all staff that could view it. The diversity of belief about the content, meaning, and use of the same datasets across this group of stakeholders cannot be explained by traditional reductionist perspectives. Traditional perspectives and methods can neither explain this situation, nor offer a way forward for the business.

### **5.3.1 The Complex Adaptive System Perspective**

A CAS perspective, where agents understand their surroundings affected by their personally accumulated experience, is able to accommodate the evidence presented here. Similarly, it is able to explain the accommodation and adjustments staff make in using these tools. Managing an organization as a CAS may offer ways forward for this particular industrial problem.

## **6. DISCUSSION**

At the Business Line level of analysis, the expectation regarding future business outcomes and how they would be realized was articulated through the business planning documentation. The paradigm that the expectations are derived from is traditional, which is unable to acknowledge that a link between cause and effect may only be knowable in hindsight, and it is unable to explain emergence, or manifestation of the 'unexpected'. In practice, the methods identified in the business plans that were expected to deliver the desired outcomes were not used, planned monitoring was not carried out.

The Business Line delivery performance displays emergent properties arising from the complexity of the technical systems under development and the actions of staff within the Business Line. A narrative that describes how and why emergent outcomes arose was provided by some of the staff members, if they had an immediate view of the key dynamics. Recognition of the CAS property that agents are not all-knowing, rather they exhibit a

bounded view and operate rationally within that sphere explains why those able to perceive and describe process narratives can do so only for the area of the organization they operate within.

The empirical research found evidence of interconnectedness, dynamics, autonomous agent behavior and bounded rationality, consistent with a CAS perspective that leads to the emergent performance of the engineering function as it works to develop Systems Engineering solutions. The emergent performance may be unexpected and dissatisfactory, since expectations are often developed from within a reductionist paradigm. The same emergent performance could be considered entirely plausible and reasonably expected if expectations were developed from a complex adaptive systems perspective.

The range of management approaches endorsed by functional and project managers also originate in a reductionist paradigm. The use of these approaches are theoretically, and as shown here in practice, ineffective in supporting pursuit of desirable outcomes from an organization that exhibits CAS properties.

The evidence began in topic 1 considering the effect that unplanned changes in staff, and work tasks had on the ability of the sprint team to deliver a completed backlog as planned. The unplanned changes emerged due to the complexity of the organization; staff are humans and are occasionally absent from work, and the short-term needs of other projects and functions in a matrix organization are deemed to take priority over the stability of the planned sprint. The theme of staff instability expanded in topic two to capture a dynamic of staff frequently changing their 'permanent' roles. In both topics 1 and 2, reductionist plans and approaches aiming to deliver projects and business plan objectives were unable to accommodate or account for the complex dynamics that were evident in the workplace. The aims that the traditionally-derived project and business plans sought to achieve were therefore jeopardized.

The third topic shows evidence of traditional views in the Business Line management. Though there is some acknowledgement that data held within the various management databases may be questionable, the managers still view the data through the databases in preference to using other methods to understand the 'remote' situation that the data nominally represents. In the case of the Key Stakeholder, in examples directly related to their work, there was some evidence of an appreciation of the role that complexity plays (in response to evidence provided in topic 2, and from their own experience of business planning in topic 3).

The Complex Adaptive System perspective used here considers agents having three levels of activity; performance, credit-assignment, and rule-discovery (Holland, 2014). For an agent within the system (for example a functional manager) to use complexity-based approaches to help manage development work towards satisfactory delivery (performance), they need to see the value and relevance of a complexity-based approach (credit-assignment), which can only be developed if familiarity with the concepts is gained (rule-discovery). Familiarity with the concepts can only be gained if the agent chooses to allocate their performance towards learning or exploring the possibilities that a complexity-based approach may offer them in pursuing the goals of their role. If a traditional paradigm is firmly engrained and strongly preferred (habitus), there would be no appetite for this learning or exploring. The 'purely-traditionally-thinking' CAS agent self-reinforces an inability to manage complexity.

A traditional view would assume some form of 'closed loop' management occurs, such as a plan-do-check-act cycle. This approach would, in theory, identify and analyze the pervasive inability of the organization to deliver as planned. In the example organization explored here, the relevant stakeholders did not engage in this aspect of organizational management. Anecdotally, the demands on staff time greatly exceed their abilities to complete their full workload, so they, as individuals, prioritize what work is actually done moment-to-moment, day-to-day. Evidence in the case study database demonstrates this point, but is not reproduced here. This dynamic is entirely understandable through a CAS lens, but not through a traditional view. This dynamic harms the self-checking mechanism inherent in traditional approaches, so not only is the emergent performance of the organization not 'as expected', but the in-built mechanisms that would be expected to raise an alarm are also disabled.

The findings from this work demonstrate that CAS dynamics can be observed within a Systems Engineering development organization, and they affect the ability to meet delivery expectations. The dominant approaches to planning and management, however, are more traditionally reductionist and deterministic in nature, and are therefore unable to understand and accommodate these dynamics in support of effective project development and desirable delivery performance. The evidence presented here show all project engineering staff sampled identify

aspects of complexity in their work. Project and product engineering staff, to varying degrees instinctively employ methods and manage expectations that complement the dynamics and inherent uncertainty faced in their work.

## 7. CONCLUSION

The research aimed to explore the question ‘how does how does complexity in the organization affect the ability of Systems Engineers to meet delivery expectations in terms of cost and time?’ The research examines three topics. Each topic demonstrates that delivery expectations are derived from perspectives that assume deterministic relations between cause and effect, which manifest through assumptions that desired outcomes can be delivered reliably and predictably through control-loop-style management activities. Each topic also demonstrated that the characteristics of specific aspects of the organization considered are complex, in the sense that cause and effect can only be understood in retrospect, so specific outcomes cannot be reliably predicted. Efforts to manage a complex adaptive system in order to enable desirable emergence could not guarantee that desired outcomes were achieved on every occasion.

The failure to meet delivery expectations in Systems Engineering development projects has been addressed in the Systems Engineering literature largely through quantitative approaches that have sought correlation between context and outcome. The research presented in this paper is concerned with causality rather than correlation. It finds that often cause and effect can only be understood in hindsight due to the complex adaptive nature of the organization, which allows us to conclude that Systems Engineering development projects carried out by complex adaptive organizations cannot be planned and controlled to guarantee that target delivery dates and costs will be achieved uniformly.

This leads us to conclude that a suitable organizational management approach to apply to a portfolio of systems engineering development projects is one that enables desirable emergence from within the complex adaptive system. Literature could not be found that examines this in practice. This management approach is a form of engagement with a wicked problem (Rittel and Webber, 1973). The reductionist view is, however, adequate for planning and managing delivery in the stable areas within the organization (Sayer, 2000:14-15), so a management approach that enables desirable emergence is suggested here as a suitable accompaniment to traditional management methods, rather than as a substitution. With the use of management approaches that support desirable emergence, comes a need to recognize that the success of delivery performance across a portfolio can not be judged solely through statistical analysis of deviations from precise, quantitative target points, such as number of days late, or the magnitude of cost over runs.

A compelling direction for further research would explore how reductionist management and CAS management approaches can complement each other in supporting Systems Engineering development organizations as they strive to meet delivery expectations. This would also support the view of Van De Ven and Poole (2005) that multiple perspectives are complementary and necessary. The industrial need is two-fold; firstly, to understand how the management approaches are relevant and can complement one another and secondly, to identify methods to supply this information and these tools to practitioners, so they - the agents comprising the system - can be more informed about how the system works, and how to enable the outcomes they desire in terms of delivery performance. Indeed, a systems engineering view of the people, tools and processes of a systems engineering development organization could be considered to shed light on how the two management approaches can applied to achieve desired performance from the organization. Complexity Leadership Theory (Uhl-Bien and Marion, 2009) also offers an explanation of how reductionist and CAS management approaches can complement each other. The theory is operationalized in two case studies described in Gilbert, et al. (2015).

## REFERENCES

- Bhaskar, R (2008) *A Realist Theory of Science*, Routledge, Oxford
- Brown, S. (2009) Naivety in Systems Engineering Research: Are we Putting the Methodological Cart before the Philosophical Horse?, 7<sup>th</sup> Annual Conference on Systems Engineering Research, Loughborough, UK.
- Edson, M. (2012) A Complex Adaptive Systems View of Resilience in a Project Team, *Systems Research and Behavioral Science*, Vol. 29, pp 499-516
- The Evidence Centre (2010) Evidence Scan: Complex Adaptive Systems, The Health Foundation, London.
- Friedman, G, and Sage, A. (2004) Case Studies of Systems Engineering and Management in Systems Acquisition, *Systems Engineering*, Vol. 7, No. 1, pp 84-96
- Gell-Mann, M. (1994) Complex Adaptive Systems, in *Complexity: Metaphors, Models, and Reality*, Eds. Cowan, G., Pines, D. and Meltzer, D., Wesley, Reading, MA. USA, pp 17-45
- Gilbert, D., Shrieves, L., Yearworth, M. (2015) A Systems Engineering Application of Complexity Leadership Theory, 6th International Conference Complex Systems Design and Management, Paris, FR.
- Gilbert, D., Yearworth, M., Oliver, L. (2014) Systems Approach to the Development and Application of Technical Metrics to Systems Engineering Projects, 12<sup>th</sup> Annual Conference on Systems Engineering Research, Redondo Beach, CA, USA.
- Hobday, M. (2000) The Project-Based Organization: An Ideal Form for Managing Complex Products and Systems?, *Research Policy*, Vol 29, pp871-893
- Holland, J. (2014) *Complexity – A Very Short Introduction*, Oxford University Press, Oxford, UK
- Kurtz, C.F., Snowden, D.J. (2003) The new dynamics of strategy: Sense-making in a complex and complicated world. *IBM Systems Journal*, Vol 42, Issue 3, pp462-483
- Martin, J., Davidz, H. (2007) Systems Engineering Case Study Development, 5<sup>th</sup> Annual Conference on Systems Engineering Research, Hoboken, NJ, USA.
- Midgley, G (2011) Theoretical Pluralism in Systemic Action Research, *Systemic Practice and Action Research*, Vol. 24, No. 1, pp 1-15
- Mingers, J. (2004) Real-izing Information Systems: Critical Realism as an Underpinning Philosophy for Information Systems, *Information and Organization*, Vol. 14, pp. 87-103
- Muller, R., Sankaran, S. and Drouin, N. (2013) Introduction in *Novel Approaches to Organizational Project Management Research*, Eds. Drouin, N., Muller, R., Sankaran, S. Copenhagen Business School Press, pp 19-33
- Palmberg, K. (2009a) Complex adaptive systems as metaphors for organizational management, *The Learning Organization*, Vol16, No. 6, pp483-498
- Palmberg, K. (2009b) Beyond Process Management – Exploring Organizational Applications and Complex Adaptive Systems, Doctoral Thesis, Lulea University of Technology, Department of Business Administration and Social Sciences
- Rittel, H. and Webber, M. (1973) Dilemmas in a General Theory of Planning, *Policy Sciences*, Vol. 4, pp155-169
- Sayer, A. (2000) *Realism and Social Science*, Sage, London.
- Schwaber, K. and Sutherland, J. (2013) The Scrum Guide: 1<sup>st</sup> ed. [ebook] Available at: <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-US.pdf#zoom=100> [Accessed 6<sup>th</sup> Jun. 2016]
- Sheard, S. (2012) Assessing the Impact of Complexity Attributes on System Development Project Outcomes, PhD Dissertation, Stevens Institute of Technology, Hoboken, NJ. USA
- Sheard, S., and Mostashari, A (2009) Principles of Complex Systems for Systems Engineering, *Systems Engineering*, Vol. 12, No. 4, pp 295-311
- Snowden, D. and Boone, M. (2007) A Leaders's Framework for Decision Making, *Harvard Business Review*, November 2007, pp 69-76
- Uhl-Bien, M., and Marion, R. (2009) Complexity Leadership in Bureaucratic forms of Organizing: a Meso Model, *The Leadership Quarterly*, Vol. 20, pp631-650
- Valerdi, R., Brown, S., Muller, G. (2010) Towards a Framework of Research Methodology Choices in Systems Engineering, 8<sup>th</sup> Annual Conference on Systems Engineering Research, Hoboken, NJ, USA.
- Valerdi, R., and Davidz, H. (2007) Empirical Research in Systems Engineering: Challenges and Opportunities of a

- New Frontier, 5<sup>th</sup> Annual Conference on Systems Engineering Research, Hoboken, NJ, USA.
- Van de Ven, A., and Poole, M. (2005) Alternative Approaches for Studying Organizational Change, *Organization Studies*, Vol. 26, No. 9, pp 1377-1404
- Van Oorschot, K. Sengupta, K., Akkermans, H., Van Wassenhove, L. (2010) Get Fat Fast: Surviving the Stage-Gate ® in NPD. *Journal of Product Innovation Management*, Vol 27, Issue 6, pp828-839
- Yin, R. (2014) *Case Study Research – Design and Methods*, 5<sup>th</sup> Edition, Sage, USA.



Table 1 Product 1 Sprint Performance

Sprint Number	Sprint Team 1		Sprint Team 2		Sprint Team 3		Sprint Team 4	
	Committed (Planned)	Completed (Actual)	Committed (Planned)	Completed (Actual)	Committed (Planned)	Completed (Actual)	Committed (Planned)	Completed (Actual)
N	11	1	53	32.5	10	10	0	0
N+1	10	3	32.5	18	0	0	0	0
N+2	10	3	45.5	23.25	15	2	9	0
N+3	13	8	45.5	19.5	24.5	5.5	19	0
N+4	5	5.5	64.5	42.25	46.5	32.5	46.5	32.5
N+5	8	6	60	22.5	22.25	4	27.25	4
N+6	0	0	56.75	23	16.5	12	22	17
Total	57	26.5	357.75	181	134.75	66	123.75	53.5
Sprint Number	Sprint Team 5		Sprint Team 6		Total			
	Committed (Planned)	Completed (Actual)	Committed (Planned)	Completed (Actual)	Committed (Planned)	Completed (Actual)		
N	29.5	16	15.5	12.5	<b>119</b>	<b>72</b>		
N+1	17.5	7	15	15	<b>75</b>	<b>43</b>		
N+2	13	5	12.5	13.25	<b>105</b>	<b>46.5</b>		
N+3	8	0	5	4	<b>115</b>	<b>37</b>		
N+4	8	0	10	4.25	<b>180.5</b>	<b>117</b>		
N+5	8.25	1	23.5	11.5	<b>149.25</b>	<b>49</b>		
N+6	16.75	6.5	23.5	4.5	<b>135.5</b>	<b>63</b>		
Total	101	35.5	105	65	<u>879.25</u>	<u>427.5</u>		

Table 2 Product 2 Sprint Performance

Sprint Number	Sprint Team 7		Sprint Team 8		Sprint Team 9		Total	
	Committed (Planned)	Completed (Actual)	Committed (Planned)	Completed (Actual)	Committed (Planned)	Completed (Actual)	Committed (Planned)	Completed (Actual)
N	125	0	115	0	0	0	<b>240</b>	<b>0</b>
N+1	318	291.5	22	0	0	0	<b>340</b>	<b>291.5</b>
N+2	1180	800	90	0	12	0	<b>1282</b>	<b>800</b>
N+3	1258.5	960	794.5	539	305	262	<b>2358</b>	<b>1761</b>
N+4	617.5	635.5	496.5	434.5	121	77	<b>1235</b>	<b>1147</b>
N+5	524	312	181	140	341	170	<b>1046</b>	<b>622</b>
N+6	35	198.5	35	194	0	4	<b>70</b>	<b>396.5</b>
Total	4058	3197.5	1734	1307.5	779	513	<u>6571</u>	<u>5018</u>

Table 3 Product 3 Sprint Performance

Sprint Number	Sprint Team 10		Sprint Team 11		Sprint Team 12	
	Committed (Planned)	Completed (Actual)	Committed (Planned)	Completed (Actual)	Committed (Planned)	Completed (Actual)
N	497.5	224.4	1363.3	1066.65	674.4	731.25
N+1	527	239.5	1493	1005.4	617.9	497.8
N+2	384.5	300.7	785.3	467.5	268	137.2
N+3	310.9	172.9	645.35	351.65	211.8	136.2
N+4	274	68.9	695.1	414.8	247.6	213.6
N+5	375.3	262.9	1029.42	498.92	375.52	163.52
N+6	322.6	230.3	1286.55	764.8	508.2	318.1
Total	<i>2691.8</i>	<i>1499.6</i>	<i>7298.02</i>	<i>4569.72</i>	<i>2903.42</i>	<i>2197.67</i>
Sprint Number	Sprint Team 13		Total			
	Committed (Planned)	Completed (Actual)	Committed (Planned)	Completed (Actual)		
N	96.2	111	<b>2133.9</b>	<b>1908.9</b>		
N+1	252.9	268.1	<b>2363.8</b>	<b>1771.3</b>		
N+2	29.6	29.6	<b>1082.9</b>	<b>634.3</b>		
N+3	68.45	42.55	<b>925.6</b>	<b>530.4</b>		
N+4	76.3	76.3	<b>1019</b>	<b>704.7</b>		
N+5	198.1	55.5	<b>1603.04</b>	<b>717.94</b>		
N+6	316.7	216.4	<b>2111.45</b>	<b>1299.3</b>		
Total	<i>12893.24</i>	<i>8266.99</i>	<u>23094.68</u>	<u>15034.38</u>		

Table 4 Sprint Summary Statistics

Product 1 (see Table 1)	R-Squared	m
Unformatted Data Points	0.771	0.509
<i>Italicised Data Points</i>	0.971	0.496
<b>Bold Data Points</b>	0.444	0.502
Product 2 (see Table 2)	R-Squared	m
Unformatted Data Points	0.916	0.740
<i>Italicised Data Points</i>	0.997	0.779
<b>Bold Data Points</b>	0.880	0.737
Product 3 (see Table 3)	R-Squared	m
Unformatted Data Points	0.884	0.656
<i>Italicised Data Points</i>	0.976	0.637
<b>Bold Data Points</b>	0.860	0.667
All Products	R-Squared	m
Unformatted Data Points	0.928	0.681

Table 5 Sprint Team Member Availability by Sprint

Product 1	Equivalent Staff Utilisation During the Sprint					
	Sprint M	Sprint M+1	Sprint M+2	Sprint M+3	Sprint M+4	Average
Staff Member A	68%	72%	56%	61%	46%	60%
Staff Member B	84%	67%	77%	81%	60%	74%
Staff Member C	89%	100%	95%	100%	67%	90%
Staff Member D	63%	95%	72%	72%	65%	73%
Staff Member E	63%	86%	77%	68%	69%	72%
Staff Member F	84%	90%	81%	86%	69%	82%
Staff Member G	89%	100%	80%	85%	70%	85%
Staff Member H			81%	88%	60%	46%
Staff Member I	84%	95%	61%	83%	57%	76%
Staff Member J				38%	29%	33%
Staff Member K					51%	51%
Staff Member L					60%	60%

Table 6 Sprint Churn by Sprint

Sprint Number	Originally Planned Work	Unplanned Staff Availability Change		Planned Work Removed	Unplanned Work Added	Abortive Work	Sprint Churn	Actual Work Delivered
	Magnitude	Magnitude	Percent	Magnitude	Magnitude	Magnitude	Percent	Magnitude
Sprint I	783	-35	-4%	12	105	0	13%	806
Sprint I+1	668	-63	-9%	267	47	-24	7%	642
Sprint I+2	465	-5	-1%	126	84	-19	18%	393
Sprint I+3	1015	-38	-4%	302	257	-58	25%	865
Sprint I+4	996	6	1%	242	283	0	28%	1088
Sprint I+5	1055	-29	-3%	292	361	-130	34%	1198
Sprint I+6	1458	-77	-5%	775	280	-161	19%	1157
Average			-4%				21%	

Table 7 Management Tools and Databases

<b>Tool</b>	<b>Contents (Units) and Scope</b>
Jira®	Work (Tasks and Problems), <i>Future, Present and History</i>
Primavera®	Work (Work packages) People (named and unnamed) <i>Future and up-to-last-report</i>
Business Planning	Backlog (£ Budget) People (Headcount) <i>Future</i>